

## Contents

<b>1 Routine/Function Prologues</b>	<b>2</b>
1.1 Fortran: Module Interface baseforcing_module.F90 (Source File: baseforcing_module.F90) . . . . .	2
1.1.1 LIS_get_base_forcing (Source File: baseforcing_module.F90) . . . . .	2
1.1.2 LIS_baseforcing_init (Source File: baseforcing_module.F90) . . . . .	2
1.1.3 forcing_init (Source File: baseforcing_module.F90) . . . . .	2
1.1.4 init_baseforcing (Source File: baseforcing_module.F90) . . . . .	3
1.1.5 get (Source File: baseforcing_module.F90) . . . . .	3
1.1.6 time_interp (Source File: baseforcing_module.F90) . . . . .	4
1.1.7 ld_getbaseforcing (Source File: baseforcing_module.F90) . . . . .	4
1.1.8 scatter_data (Source File: baseforcing_module.F90) . . . . .	5

## 1 Routine/Function Prologues

### 1.1 Fortran: Module Interface baseforcing\_module.F90 (Source File: baseforcing\_module.F90)

This module contains interfaces and subroutines that controls the incorporation of model forcing

#### REVISION HISTORY:

14Nov02 Sujay Kumar Initial Specification

---

#### 1.1.1 LIS\_get\_base\_forcing (Source File: baseforcing\_module.F90)

##### INTERFACE:

```
interface LIS_get_base_forcing
    module procedure ld_getbaseforcing
end interface
```

---

#### 1.1.2 LIS\_baseforcing\_init (Source File: baseforcing\_module.F90)

##### INTERFACE:

```
interface LIS_baseforcing_init
    module procedure init_baseforcing
end interface
```

---

#### 1.1.3 forcing\_init (Source File: baseforcing\_module.F90)

Sets up functions for reading model forcing

##### INTERFACE:

```
subroutine forcing_init()
```

##### USES:

```
use baseforcing_pluginMod
```

---

#### 1.1.4 init\_baseforcing (Source File: baseforcing\_module.F90)

Initializes model forcing variables and allocates memory

INTERFACE:

```
subroutine init_baseforcing()
```

USES:

```
use def_ipMod, only: def_ip_input, allocate_ip, &
    def_budgetip_input
use lisdrv_module, only: lis, getforcing
#ifndef ( defined OPENDAP )
    use opendap_module
#endif
integer :: kgdsi(200)
```

CONTENTS:

```
kgdsi = 0
call forcing_init()
call allocate_forcing_mem()
#ifndef ( defined OPENDAP )
    call defnates(getforcing(),kgdsi)
    call define_gds(lis)
    call allocate_ip(parm_nc*parm_nr)
    call def_ip_input(kgdsi)
#else
    if( masterproc) then
        call defnates(getforcing(),kgdsi)
        call allocate_ip(lis%d%lnc*lis%d%lnr)
        call def_ip_input(kgdsi)
    !       call def_budgetip_input(kgdsi)
    endif
#endif
```

#### 1.1.5 get (Source File: baseforcing\_module.F90)

Retrieves and interpolates model forcing

INTERFACE:

```
subroutine get()
```

USES:

```
use lisdrv_module, only: getforcing
```

CONTENTS:

```
call getf(getforcing())
```

**1.1.6 time\_interp (Source File: baseforcing\_module.F90)**

Computes temporal interpolation

INTERFACE:

```
subroutine time_interp()
```

*USES:*

```
use lisdrv_module,only :getforcing
```

CONTENTS:

```
call timeinterp(getforcing())
```

CONTENTS:

```
nmif = getnmif()
if(masterproc) then
    allocate(glbdata1(nmif,lis%d%glbngrid))
    allocate(glbdata2(nmif,lis%d%glbngrid))
else
    allocate(glbdata1(nmif,gdi(iam)))
    allocate(glbdata2(nmif,gdi(iam)))
endif
```

---

**1.1.7 ld\_getbaseforcing (Source File: baseforcing\_module.F90)**

Retrieves model forcing and invokes spatial and interpolation routines

INTERFACE:

```
subroutine ld_getbaseforcing()
```

*USES:*

```
use lisdrv_module, only: lis
```

CONTENTS:

```
#if ( defined OPENDAP )
    call get()
#else
    if ( masterproc ) then
        call get()
    endif
    call MPI_BCAST(lis%f%findtime1,1,MPI_INTEGER,0, &
                  MPI_COMM_WORLD, ier)
    call MPI_BCAST(lis%f%findtime2,1,MPI_INTEGER,0, &
```

```
    MPI_COMM_WORLD, ier)
if(lis%f%findtime1 ==1 .or.  &
   lis%f%findtime2==1) then
  if(npes > 1) then
    call scatter_data()
  endif
endif
#endif
call time_interp()
```

---

### 1.1.8 scatter\_data (Source File: baseforcing\_module.F90)

Distributes the forcing data on compute nodes

INTERFACE:

```
subroutine scatter_data()
```

USES:

```
use lisdrv_module, only : lis
```